

Lab 6

Data Types and Math Operators

GOALS:

1. Students will be able to create a program that input values from the keyboard.
2. Students will be able to store values that are inputted from the keyboard into memory locations.

ESSENTIAL QUESTIONS

1. What is the purpose of the TextBox?
2. What is the purpose of data types?

Data Types

Data types are built-in meaning they pre-exist. It allows us to give meaningful names to memory locations. You have several types of data types. Listed below are 7 data types that Visual Basic supports.

Type	Prefix	Represents	Bytes
Single	sgl	Numbers with decimals	4
Double	dbl	Numbers with decimals	8
Integer	int	Numbers without decimals	2
Long	lng	Numbers without decimals	4
Currency	cur	Numbers representing dollar amounts	8
String	str	Set of characters	1 byte per character
Boolean	bln	True or False	2

To create memory locations called **variables**, you must use the **Dim** statement. The **Dim** statement stands for **dimension** which creates a memory location and the Data Type determines what type of values will be stored in it.

Example:

Dim intNum as Integer 'creates a variable called intNum that can only store integers
Form

Dim prefixOfDataType_Name **As** DataType

Rules for naming variables

1. Must start with a letter
2. Must contain only letters, digits, and underscore (_) character. Periods, spaces, and other special characters are not allowed.
3. Cannot exceed 255 characters

Variables are not case sensitive in Visual Basic. This mean you can spell the variables in two different cases and visual basic will consider it the same word.

TextBox

TextBoxes allows us to input information from the keyboard into our program. All information inputted in a TextBox whether it is numerical or text comes in as text. It gets transform to the appropriate type when it gets stored in memory. This process is called **automatic type conversion**. To retrieve values from the TextBox, you must set a memory location equal to the TextBox.

Example:

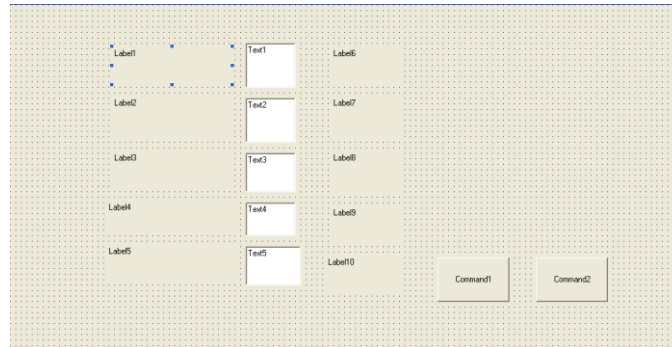
```
intNum = txtInput .text
```

The above statement takes the value typed into the TextBox and stores it into the memory called intNum. It uses the assignment operator to do this.

Assignment Operator (=)

The equal symbol is called the **assignment operator**. It takes what is on the right side of it and stores it on the left side. So in the example above it took whatever was in the txtInput.text object and stored it in intNum.

Step 1: Create the following form (5 text boxes, 10 labels, 2 buttons)



Step 2: Give the objects the following names

Command1 → cmdDone

Command2 → cmdOutput

Label1 → lblInt

Label2 → lblDouble

Label3 → lblBoolean

Label4 → lblSingle

Label5 → lblLong

Label6 → lblOutInt

Label7 → lblOutDouble

Label8 → lblOutBoolean

Label9 → lblOutSingle

Label10 → lblOutSentence

Text1 → txtInt

Text2 → txtDouble

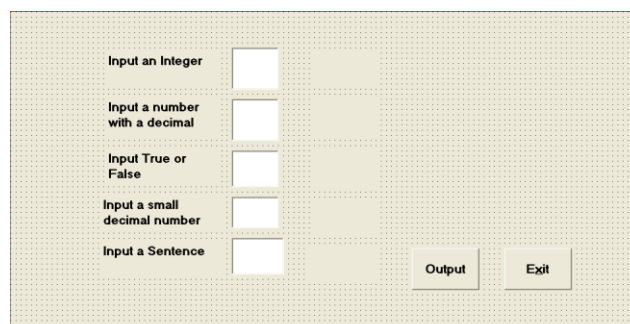
Text3 → txtBoolean

Text4 → txtSingle

Text5 → txtSentence

Form → frmLab6

Step 3: Give the command buttons & labels the following captions:



Step 4: Code the Output button

```
Private Sub cmdOutput_Click()  
    Dim intVal As Integer  
    Dim dblNum As Double  
    Dim blnAns As Boolean           'Creates memory locations with the following data types  
    Dim sglVal As Single  
    Dim strSent As String  
  
    intVal = txtInt.Text  
    dblNum = txtDouble.Text  
    blnAns = txtBoolean.Text       'Reads values from text box and store it in the appropriate memory location  
    sglVal = txtSingle.Text  
    strSent = txtSentence.Text  
  
    lblOutInt.Caption = intVal  
    lblOutDouble.Caption = dblNum  
    lblOutBoolean.Caption = blnAns  
    lblOutSingle.Caption = sglVal  
    lblOutSentence.Caption = strSent  
  
End Sub
```

Step 5: Code the Exit button. Don't forget the shortcut key

Step 6: Save and Run the program

Questions:

1. What is the highest value that you can store in an Integer location? Try by entering values in textbox?
2. What happens if you type the wrong type of value in a TextBox?
3. Can you declare more than one variable on the same line?
4. Can you create two memory locations with the exact same name but different data types?

PART 2

MATH OPERATORS

Visual Basic supports many different math operators.

Math Operators:

1. ^ → exponents
2. () → parenthesis
3. * → multiplication
4. / → real division
5. \ → Integer division
6. Mod → Modulus
7. + → Addition
8. - → Subtraction

Order of precedence also known as Order of operations is basically the same as Algebra.

1. $()$, $^$
2. $*$, $/$ from left to right
3. \backslash ,
4. Mod
5. $+$, $-$ left to right

\backslash - Integer Division – truncates the decimal portion of the quotient which results in an Integer.

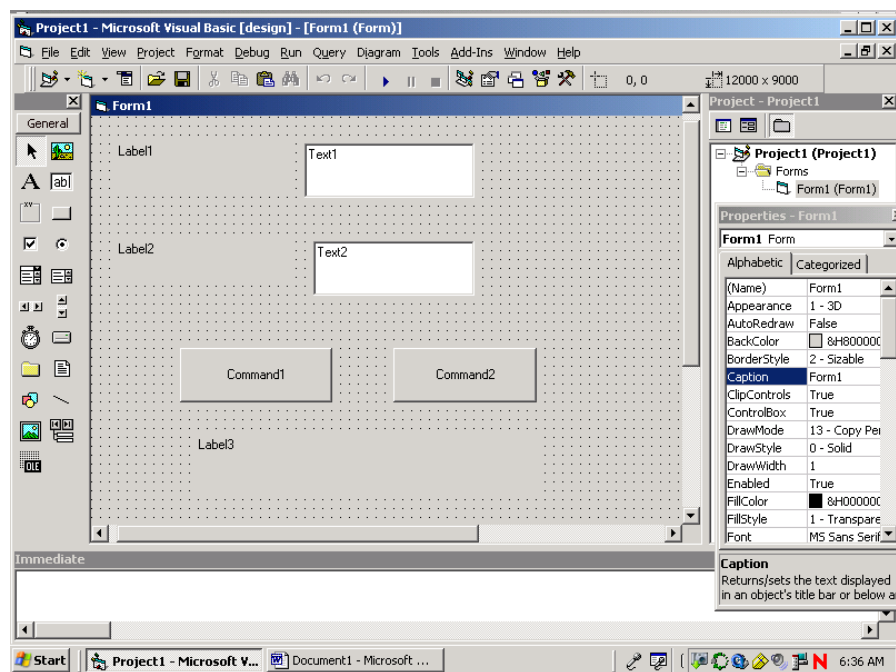
Example: $\text{intSum} = 20 \backslash 7$ 'intSum is assigned 2

Mod – Modulus Division – returns the remainder resulting from division.

Example: $\text{intSum} = 20 \text{ Mod } 7$ 'intSum is assigned 6

Part 2 Objective: You will be able to input two values and perform a math calculation on the two values when a button is clicked. The result will also be displayed to the screen.

Step 1: Design a form similar to the one below:

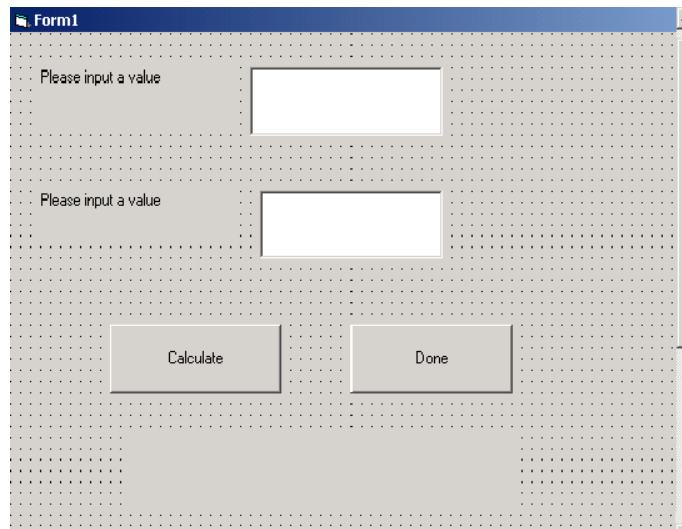


Notice you have 3 labels, 2 command buttons, and 2 text boxes.

Step 2: Give each label, button and text box an appropriate name using the prefixes.

Label1 = lblDisplay1
Label2 = lblDisplay2
Label3 = lblDisplayAns
Command1 = cmdCalculate
Command2 = cmdDone
Text1 = txtInput1
Text2 = txtInput2
Form = frmLab2

Step 3: Remove all the display labels and put the appropriate messages in the proper places used the following as a guide.

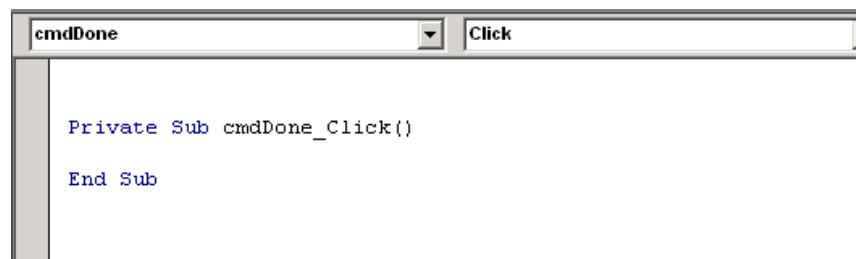


Notice that the text boxes are blank and the bottom label is blank

Step 4:

Open the code window.

Display the command button for cmdDone by clicking on the down arrow by the word General.

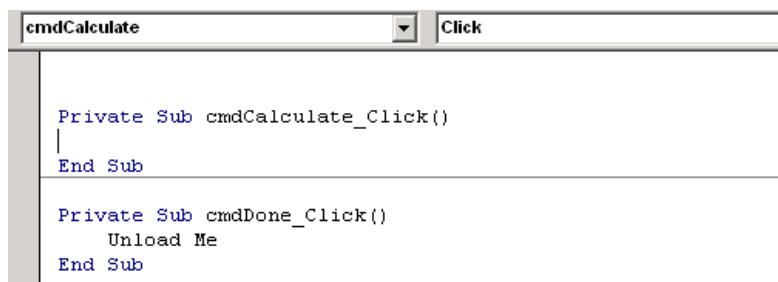


Type between Sub and End Sub the following:

Unload Me or Unload frmLab2

Step 5:

Next display the cmdCalculate code by clicking the down arrow and selecting cmdCalculate.



Between the Sub and End Sub type the following:

Dim intVal1 As Integer

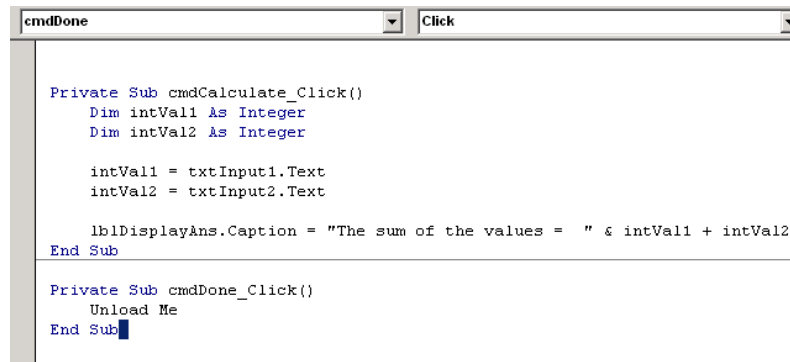
Dim intVal2 As Integer

intVal1 = txtInput1.Text

intVal2 = txtInput2.Text

lblDisplayAns.Caption = "The sum of the values = " & intVal1 + intVal2

When completed it should look like the following:



```
Private Sub cmdCalculate_Click()  
    Dim intVal1 As Integer  
    Dim intVal2 As Integer  
  
    intVal1 = txtInput1.Text  
    intVal2 = txtInput2.Text  
  
    lblDisplayAns.Caption = "The sum of the values = " & intVal1 + intVal2  
End Sub  
  
Private Sub cmdDone_Click()  
    Unload Me  
End Sub
```

Compile and Run program

Project 1:

Try the same thing with other math operations. Use the following and try to determine what each one is:

*

-

/

\

^

Mod

Identify each of the math operators listed above.

Questions

1. Why do you think it is important to identify all objects with a prefix?

2. What do you think is the purpose of the following statements?

Dim intVal1 As Integer

Dim intVal2 As Integer

3. What is the purpose of the statements?

intVal1 = txtInput1.Text

intVal2 = txtInput2.Text

4. What is the purpose of the symbol & in the statement **"lblDisplayAns.Caption = "The sum of the values =" & intVal1 + intVal2"** ?